CopyCheck: 한국어 표절 검사 시스템

장은서[O], 권도형, 김낙원, 박소영, 강승식
국민대학교 컴퓨터공학부
akdangz@kookmin.ac.kr, kdhlook@naver.com, nwkim12@naver.com,
park-soyeong@nate.com, sskang@kookmin.ac.kr

CopyCheck: Korean Plagiarism Detection System

Eun-seo Jang[O], Do-Hyoung Kwon, Nak-Won Kim, So-Yeong Park, Seung-Shik Kang
School of Computer Science, Kookmin University

요 약

기존의 표절 검사 소프트웨어의 경우에는 수행 시간이 지나치게 오래 걸리거나 표절의 의미가 희박한 구간들을 찾는 등의 문제가 있었다. 본 논문은 대학에서 과제물 표절 검사에 활용할 수 있는 소프트웨어인 CopyCheck을 설계 및 개발하였다. CopyCheck은 각각의 대상 문서로부터 문서 고유의 시그니처 세트를 추출·비교하여 표절이 의심되는 문서들 간의 중복 인덱스 세트를 만들어 의심 구간들을 추려낸 다음 지역 정렬 방법을 이용하여 일치 구간을 찾아내는 방법으로 많은 문서들을 대상으로도 표절 구간들을 빠르게 찾아낸다.

주제어: 표절 검사, 문서 시그니처 세트, 중복 인덱싱, 지역 정렬

1. 서론

최근 들어 대학가의 논문 혹은 과제물의 표절에 관한 문제가 사회적 관심을 받고 있다. 동시에 이러한 표절 행위를 막기 위한 노력도 끊임없이 이루어지고 있다. 하지만 이러한 표절 행위를 수작업으로 검사하는 데에 소요되는 인적·시간적 비용이 너무 큰 것이 사실이다. 이러한 이유로 컴퓨터를 활용한 표절 검사 소프트웨어에 대한 연구가 활발히 이루어지고 있으며 이미 사용 가능한 소프트웨어도 다수 공개된 상태이지만, 검사 시간이 지나치게 오래 걸리거나 표절의 의미가 희박한 구간을 찾는 등의 문제가 있다.

본 논문은 검사 대상 문서 군에서 표절이 의심되는 문서 군을 군집화하고, 의심 문서들 간의 문자열 일치구간을 빠르고 정확하게 찾아낼 수 있는 표절 검사 소프트웨어인 CopyCheck을 설계 및 개발하였다. 표절 의심 문서를 찾아내기 위하여 문서 시그니처 세트를 추출·비교하는 방법을 사용하였으며, 실제 문서를 비교하는 데에 중복 인덱스 방법과 지역 정렬법을 사용하였다.

통상적으로 대상 문서들 간에 6어절 이상이 일치하는 경우 표절이라고 판단한다. 하지만 일부 검사 소프트웨어들이 이러한 정책을 설정하지 않아 표절의 의미가 희박한 1어절 또는 2어절의 구간을 찾는 경우가 있었다. 해외에서는 이미 다양한 표절 검사 솔루션들이 개발되었다. 하지만 대부분이 영문 검사만을 지원하여 한글 문서를 처리하기에는 부적절한 경우가 대부분이다.

휴먼토크의 Anti-Piracy는 다양한 문서 포맷을 지원하고 검사 수행 속도가 빠르며 텍스트 이외에도 이미지, 표 등을 검사할 수 있다는 특징이 있지만 사용자에게 제공하는 정보가 많지 않고 검사 결과에서 원문이 올바르게 출력되지 않는 문제가 발생하는 경우가 있었다.[1] 부산대학교에서 개발한 소프트웨어인 DeVAC은 표절 검사 결과를 다양하게 출력해 주지만 표절 검사를 서버에서 처리하기 때문에 파일 업로드와 결과 다운로드에 추가적인 작업이 필요하다. 또한 표절 검사를 의뢰하는 사용자의 수, 검사 대상 문서의 개수·크기가 표절 검사의 속도에 부정적인 영향을 준다.[2]

슨보소프트에서 개발한 COPYLESS는 표절 검사에 웹 검색을 활용하고 그 결과를 HTML문서 형태의 보고서로 출력해준다. 하지만 웹 검색에 오랜 시간을 소요하며 종종 표절 여부와는 거리가 있는 결과를 산출한다.[3]

2. CopyCheck

우리는 서론에서 소개한 기존 소프트웨어들의 단점을 극복하고자 새로운 표절 검사 소프트웨어인 CopyCheck를 설계 및 개발하였다.

2.1 문서 시그니처 세트

다수의 문서를 대상으로 표절 검사를 수행하기 위한 전처리 과정으로 표절이 의심되는 문서들을 군집화하는 작업이 필요하다. 본 논문은 이 작업을 문서의 시그니처 세트를 생성 및 비교하는 방법으로 처리하였다.

시그니처 세트는 분할, 선택, MD5 암호화의 세 단계를 거쳐 생성된다.[4] 각 문서는 Hashed-Breakpoint 분할법을 이용하여 여러 개의 문자열 조각들로 분할된다. 이 분할법은 문서에서 단어를 하나씩 가져와 조각에 넣고 단어를 인자로 하는 문자열 해시 함수를 호출한다. 만약 호출 결과값을 상수로 나눈 나머지가 0이 될 때마다 새로운 조각을 생성하는 방법이다.[5]

Hashed-Breakpoint 분할법은 문서를 동일한 조건으로 분할하기 때문에 대상 문서들에서 동시에 등장한 단어에 대한 결과 값이 0이 되어 분할되는 경우 일치 구간을 찾을 수 있는 확률이 높다는 특징이 있다.

<표 1> Hashed-Breakpoint 분할법 예시

문서 A	...최근 들어 논문 표절에 대한 관심이...
문서 B	...최근에 논문 표절에 대한 관심이...
hash values	h("논문")%C = 0 h("관심이")%C = 0
조각 A	논문 표절에 대한 관심이
조각 B	논문 표절에 대한 관심이

분할이 완료된 문서 조각들은 조각의 크기를 기준으로 선택하여 시그니처 세트들을 만들게 된다. 조각의 크기는 단어의 개수를 사용하며, 표절 기준 크기인 6부터 거리가 가까운 $2\sqrt{n}$개 (n= 조각의 개수) 를 선택한다. 선택된 조각들은 MD5 암호화를 거쳐 일정 길이의 문자열로 변환되며 이 문자열 세트를 문서 하나의 시그니처 세트라고 한다. 검사 대상 문서들마다 이러한 시그니처 세트를 생성하여 표절 검사 대상을 추려내게 된다.

2.2 중복 인덱스 세트

표절이 의심되는 두 문서의 모든 구간을 검사하는 방법은 너무 많은 시간이 소요되기 때문에 문서를 색인한 인덱스 파일을 생성하여 활용한다. 본 논문에서 설계한 소프트웨어는 문서 별로 인덱스 파일을 생성하는 것이 아니라 두 문서에서 동일하게 발견되는 색인어와 해당 오프셋만을 기록하는 중복 인덱스 파일을 생성한다.

색인어 생성 정책은 각 단어의 첫 번째 오프셋부터 10음절씩 잘라 생성하는 방법을 사용하였다. 색인어 뒤에는 각 문서에서 색인어가 등장한 오프셋 정보가 저장되며, 이러한 정보들은 실제 일치 구간을 찾는 작업인 지역 정렬의 시작 위치를 결정하는 역할을 하게 된다.

<표 2> 색인어 생성 정책 예시

본 논문에서 설계한 소프트웨어는 문서 별로 인덱스 파일을 만드는 것이 아니라 두 문서에서 동일하게 발견되는 인덱스와 해당 오프셋만을…	
색인어 1	본논문에서설계한소프
색인어 2	논문에서설계한소프트
색인어 3	설계한소프트웨어는문

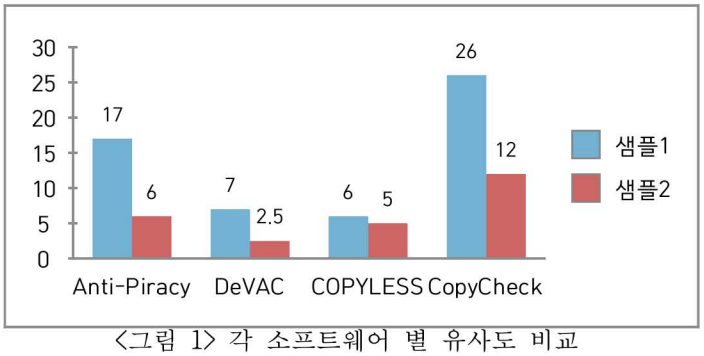
2.3 지역 정렬법(Local Alignment)

지역 정렬법은 본래 DNA 염기 서열의 유사도를 검사하는 방법이다. 이를 문자열 대조에 활용하면 유사도와 함께 주어진 두 문자열 간에 일치하는 구간 또한 찾아낼 수 있어 표절 검사에 유용하다. CopyCheck은 지역정렬 수행을 위하여 Smith-Waterman 알고리즘을 사용한다.[6]

지역 정렬은 중복 인덱스 파일을 읽어 들여 해당 오프셋에서부터 일정 길이의 문자열을 가져와 수행된다. 일치 구간의 시작과 끝 오프셋과 지역 정렬에 사용된 문자열들의 유사도를 결과 값으로 돌려주며, 이를 일정 유사도 수치로 걸러내어 표절로서의 의미가 희박한 결과들을 제거한다.

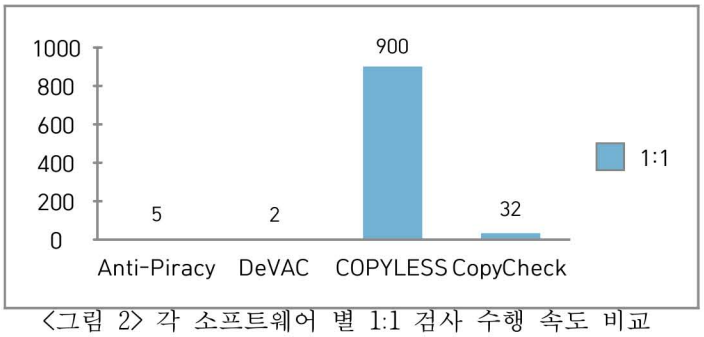
3. 결론

CopyCheck의 시그니처 세트 기법이 표절로 의심되는 문서들을 잘 찾아낼 수 있는지 시험하였다. 서로 관련이 없는 문서군에 표절 의심 문서 두 개를 포함하여 검사를 수행한 결과 두 문서를 표절 의심으로 판정하였다. 또한 동일 주제의 과제물 10개로 구성된 문서군에서 4개 문서의 일부를 복사한 문서를 포함하여 시그니처 세트를 대조한 결과, 2개의 문서를 표절로 판단하였다. 그림 1은 국내에 출시된 표절 검사 소프트웨어와 유사도 측정 결과를 비교한 것이다.[1,2,3]



<그림 1> 각 소프트웨어 별 유사도 비교

그림 2에서는 국내에 출시된 표절 검사 소프트웨어의 수행 속도를 비교하였다. 각 소프트웨어 별로 동일한 샘플 논문 두 편을 1:1 비교한 결과를 측정하였다.[1,2,3]



<그림 2> 각 소프트웨어 별 1:1 검사 수행 속도 비교

참고문헌

[1] 휴먼토크, Anti-Privacy, http://www.hmtalk.com/anti/anti06.php (2012-09-10).

[2] 류창건 , 김형준 , 조환규, "한글 말뭉치를 이용한 한글 표절 탐색 모델 개발", 정보과학회논문지: 컴퓨팅의 실제 및 레터 제14권 제2호, pp.231-235, 2008.

[3] SnboSoft, COPYLESS, http://snboard.mireene.com/snbosoft/2-3.html (2012-09-10).

[4] Raphael A. F & Arkady Z & Krisztian M & Heinz S, "Signatures extraction for overlap detection in documents", Australian Computer Science Communications, Vol 24, Issue 1, pp.59-64, 2002.

[5] Narayanan S, Hector G, "Building a scalable and accurate copy detection mechanism", Proceedings of the first ACM international conference on Digital libraries, pp.160-168, 1996.

[6] Smith T.F, Waterman M.S, "Identification of common molecular subsequences", Journal of molecular biology, Vol 147, pp.195-197, 1981.